CSE520: Computational Geometry
Lecture 16
The Lifting Map

Antoine Vigneron

Ulsan National Institute of Science and Technology

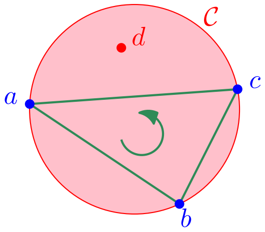
May 6, 2020

Outline

- Today, I will present a simple $O(n^2)$ time algorithm for computing the Delaunay triangulation.
- Before this, I will introduce the *lifting map*, which gives a different view of the Delaunay triangulation. It will be needed to analyze the algorithm.

- References:
- Textbook Chapter 9.
- Dave Mount's lecture notes, Lecture 18.
- H. Edelsbrunner's book *Geometry and topology of mesh generation*, Chapter 1.

InCircle Test



$\mathrm{inCircle}(a, b, c, d) < 0$

Definition (InCircle test)

Given a counterclockwise triangle abc, with circumcircle $\mathcal{C}$, the inCircle test returns a value such that:

- $\mathrm{inCircle}(a, b, c, d) = 0$ if $d \in \mathcal{C}$,
- $\mathrm{inCircle}(a, b, c, d) > 0$ if d is outside $\mathcal{C}$, and
- $\mathrm{inCircle}(a, b, c, d) < 0$ if d is inside $\mathcal{C}$.

Expression

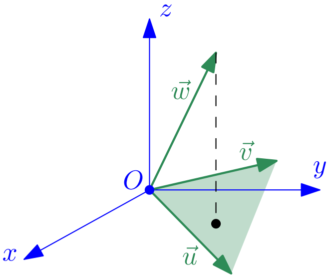
- We can use the following expression:

$$\mathrm{inCircle}(a, b, c, d) = \det \begin{pmatrix} 1 & a_x & a_y & {a_x}^2 + {a_y}^2 \\ 1 & b_x & b_y & {b_x}^2 + {b_y}^2 \\ 1 & c_x & c_y & {c_x}^2 + {c_y}^2 \\ 1 & d_x & d_y & {d_x}^2 + {d_y}^2 \end{pmatrix}.$$

- Why does it work?
- Next 10 slides: Geometric proof.

- D. Mount's notes 18: Different proof, through algebra.
- I reversed the sign of $\mathrm{inCircle}(\cdot)$ compared with D. Mount's notes, in order to simplify the presentation.

Orientation of a Triple of Vectors in $\mathbb{R}^3$



$\text{orientation}(\vec{u},\vec{v},\vec{w}) > 0$

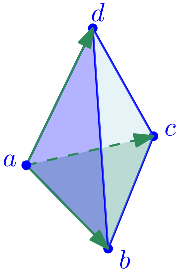
$\text{orientation}(\vec{v},\vec{u},\vec{w}) < 0$

(right thum rule)

- The orientation of a triple of vectors $(\overrightarrow{u}, \overrightarrow{v}, \overrightarrow{w})$ is given by the sign of:

$$\det \begin{pmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{pmatrix}$$

Orientation of a Tetrahedron



$\text{orientation}(abcd) = \text{orientation}(\overrightarrow{ab}, \overrightarrow{ac}, \overrightarrow{ad}) > 0$

- The orientation of tetrahedron *abcd* is the orientation of $(\overrightarrow{ab}, \overrightarrow{ac}, \overrightarrow{ad})$.

Orientation of a Tetrahedron

- By developing with respect to first column, we obtain

$$\begin{aligned}
\text{orientation}(abcd) &= \det \begin{pmatrix} b_x - a_x & b_y - a_y & b_z - a_z \\ c_x - a_x & c_y - a_y & c_z - a_z \\ d_x - a_x & d_y - a_y & d_z - a_z \end{pmatrix} \\
&= \det \begin{pmatrix} 1 & a_x & a_y & a_z \\ 0 & b_x - a_x & b_y - a_y & b_z - a_z \\ 0 & c_x - a_x & c_y - a_y & c_z - a_z \\ 0 & d_x - a_x & d_y - a_y & d_z - a_z \end{pmatrix}.
\end{aligned}$$

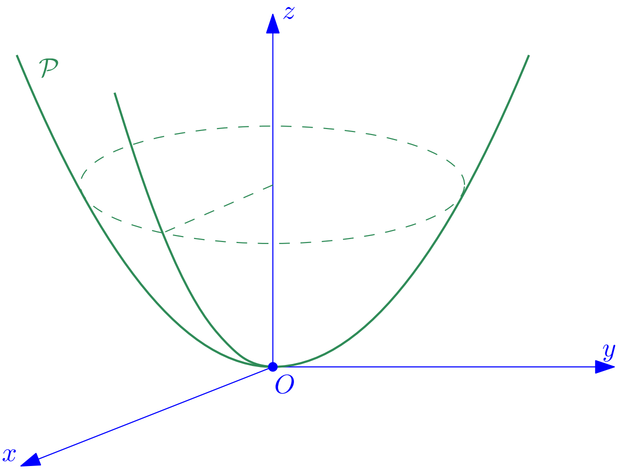
Orientation of a Tetrahedron

- Now we add the first row to rows 2–4:

$$\text{orientation}(abcd) = \det \begin{pmatrix} 1 & a_x & a_y & a_z \\ 1 & b_x & b_y & b_z \\ 1 & c_x & c_y & c_z \\ 1 & d_x & d_y & d_z \end{pmatrix}.$$

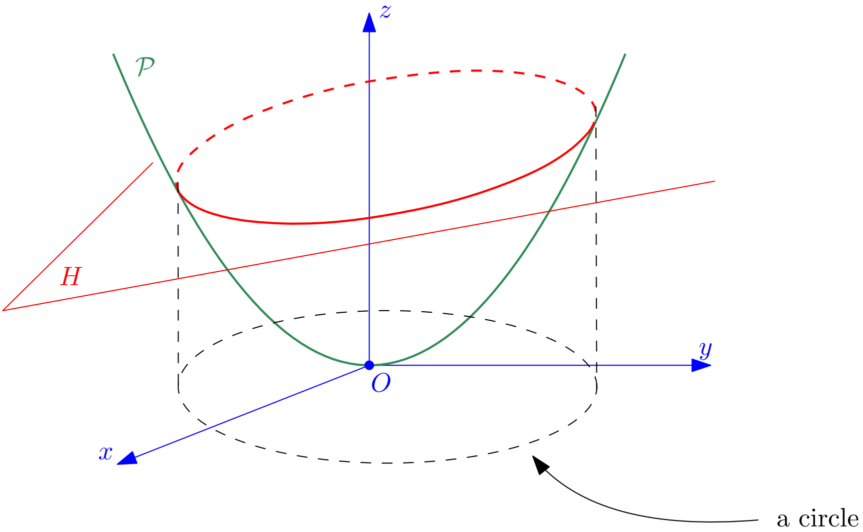
- It generalizes the 2D-counterclockwise (CCW) predicate from Lecture 2.

Paraboloid $\mathcal{P}$



- Let $\mathcal{P}$ be the paraboloid with equation $z = x^2 + y^2$ in $\mathbb{R}^3$.

Paraboloid $\mathcal{P}$



Paraboloid $\mathcal{P}$

Property

When H is a non-vertical plane, the projection of $H \cap \mathcal{P}$ onto plane Oxy is a circle.

Proof.

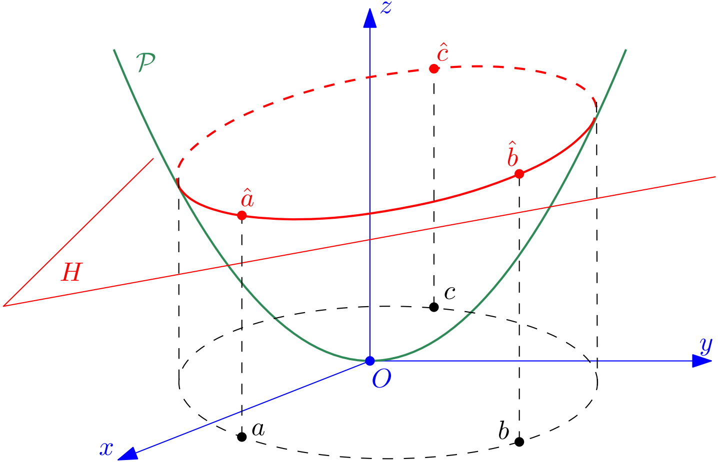
The plane H has equation $z = \alpha x + \beta y + \gamma$, so its projection $H \cap \mathcal{P}$ onto the horizontal plane Oxy has equation $x^2 + y^2 = \alpha x + \beta y + \gamma$. □

The Lifting Map

Definition (Lifting map)

The lifting map is the vertical projection of the horizontal plane onto the paraboloid $\mathcal{P}$: Any point $p = (p_x, p_y)$ is mapped to the point $\hat{p} = (p_x, p_y, p_x^2 + p_y^2)$.

Proof of the InCircle Test Expression



Proof of the InCircle Test Expression

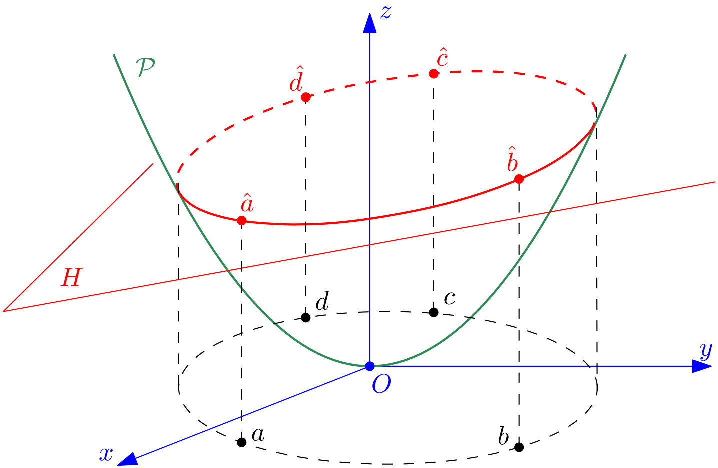
- We lift a, b, c and d:

$$\begin{aligned}\hat{a} &= (a_x, a_y, a_x^2 + a_y^2) \\ \hat{b} &= (b_x, b_y, b_x^2 + b_y^2) \\ \hat{c} &= (c_x, c_y, c_x^2 + c_y^2) \\ \hat{d} &= (d_x, d_y, d_x^2 + d_y^2)\end{aligned}$$

- We denote by H the plane through $\{\hat{a}, \hat{b}, \hat{c}\}$
- $\mathrm{inCircle}(a, b, c, d) = 0$ means that orientation$(\hat{a}, \hat{b}, \hat{c}, \hat{d}) = 0$.
- So $\hat{d} \in H$, and thus $d \in \mathcal{C}$.

Proof: First Case

- We just proved that a, b, c, d are cocircular iff $\hat{a}, \hat{b}, \hat{c}, \hat{d}$ are coplanar.



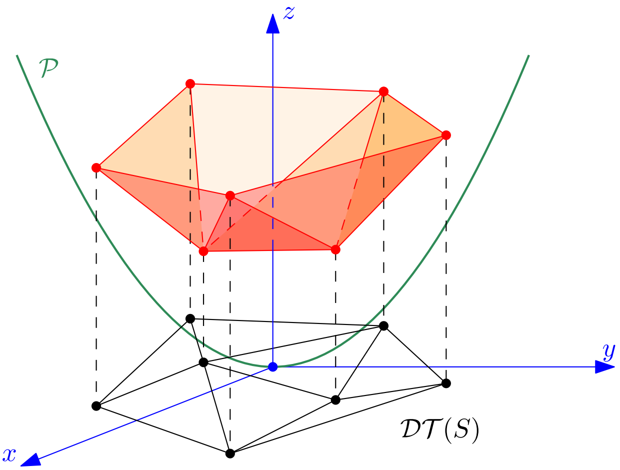
Proof: Remaining Cases

- Assume that $CCW(a, b, c) > 0$.

- $\mathrm{inCircle}(a, b, c, d) > 0$ means that Orientation$(\hat{a}, \hat{b}, \hat{c}, \hat{d}) > 0$.
- Then $\hat{d}$ is above H.
- So d is outside the circumcircle of abc.

- $\mathrm{inCircle}(a, b, c, d) < 0$ means that Orientation$(\hat{a}, \hat{b}, \hat{c}, \hat{d}) < 0$.
- Then $\hat{d}$ is below H.
- So d is inside the circumcircle of abc.

New Interpretation of the Delaunay Triangulation



- We lift the Delaunay triangulation onto the paraboloid.

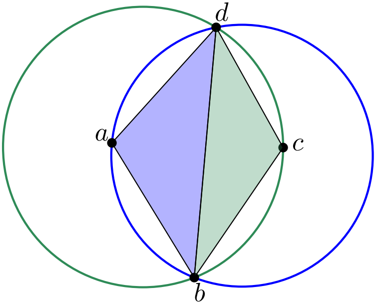
Circumcircle Property

- $S = \{s_1, s_2, \dots s_n\}$ is a set of points in the plane in general position.
- We denote $\hat{S} = \{\hat{s}_1, \hat{s}_2, \dots \hat{s}_n\}$.
- Previous lecture: The triangle $s_i s_j s_k$ is a face of $\mathcal{DT}(S)$ iff its circumcircle is empty.
- It means that $\hat{s}$ is above the plane through $\hat{s}_i \hat{s}_j \hat{s}_k$ for every $s \in P \setminus \{s_i, s_j, s_k\}$.
- In other words, $\hat{s}_i \hat{s}_j \hat{s}_k$ is a facet of the lower hull of $\hat{P}$.

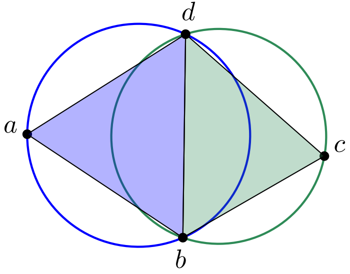
Theorem

$\mathcal{DT}(S)$ is the projection of the edges of the lower hull of $\hat{S}$ onto the plane $z = 0$.

Edge Flip



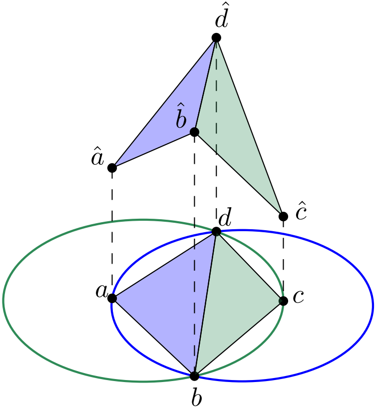
or



Property

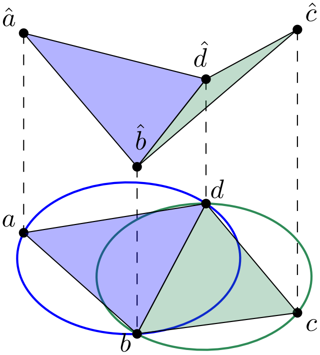
Let $abcd$ be a convex quadrilateral. Then either c is inside the circumcircle of abd and a is inside the circumcircle of bcd, or c is outside circumcircle of abd and a is outside the circumcircle of bcd.

Proof



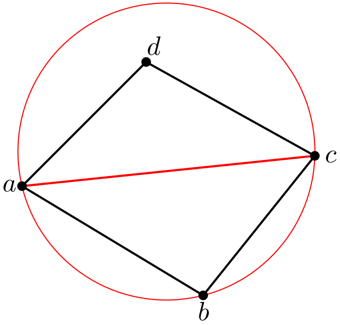
concave

or

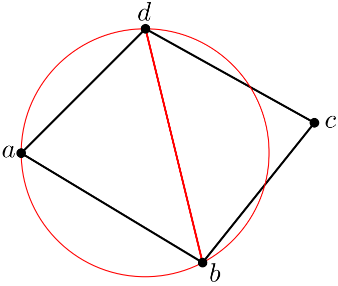


convex

Edge Flip: Definition



ac is illegal



bd is locally Delaunay

Definitions

- Let S be a set of n points in $\mathbb{R}^2$. We say that S is in *general position* if no 4 points in S are cocircular.
- Let $\mathcal{T}$ be a triangulation of S. Suppose that $abcd$ is a convex quadrilateral, such that acb and acd are faces of $\mathcal{T}$.

Definition (Locally Delaunay edge)

The edge ac is locally Delaunay iff d is outside the circumcircle of abc.

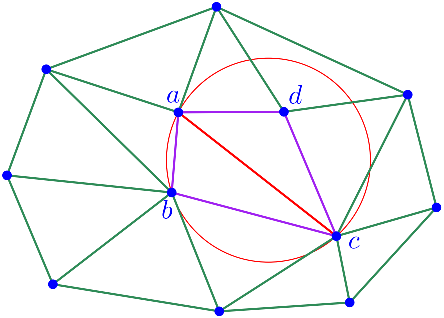
Definition (Illegal edge)

The edge ac is illegal iff d is inside the circumcircle of abc.

- We can decide whether ab is locally Delaunay or illegal by computing the sign of $CCW(a, b, c)$ and the sign of $\mathrm{inCircle}(a, b, c, d)$.

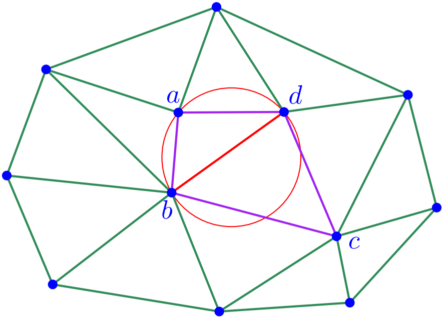
Edge Flip

If ac is illegal, we can perform an *edge flip*: Remove ac from $\mathcal{T}$ and insert bd.



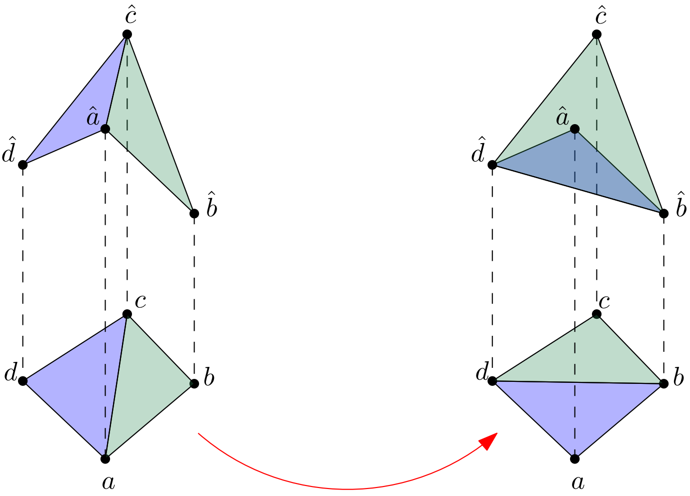
Edge Flip

If *ac* is illegal, we can perform an *edge flip*: Remove *ac* from $\mathcal{T}$ and insert *bd*.



Now *bd* is locally Delaunay.

Edge Flip: Interpretation



- The lifted triangulation gets lower, and becomes convex.

Edge Flip

Theorem

Let $\mathcal{T}$ be a triangulation of S. Then $\mathcal{T} = \mathcal{DT}(S)$ iff all the edges of $\mathcal{T}$ are locally Delaunay.

Proof.

- If $\mathcal{T}$ is Delaunay, then clearly all edges are locally Delaunay.
- Other direction:
  The triangulation is locally Delaunay
  $\Leftrightarrow$ the lifted triangulation is *locally* convex
  $\Leftrightarrow$ the lifted triangulation is *globally* convex
  $\Leftrightarrow$ the triangulation is (globally) Delaunay.

$\square$

A First Algorithm

Idea:

- Start with an arbitrary triangulation $\mathcal{T}$ of S.
- If all the edges of $\mathcal{T}$ are locally Delaunay, then we are done.
- Otherwise, pick an illegal edge and flip it.
- Repeat this process until each edge is locally Delaunay.

We will use a stack. Invariants:

- All the illegal edges are in the stack.
  - But some locally Delaunay edges may be in the stack too.
- The edges stored in the stack are *marked*, the others are not.
  - We use it to avoid having several copies of the same edge in the stack.

A First Algorithm

Pseudocode

procedure SLOWDELAUNAY(S)
  compute a triangulation $\mathcal{T}$ of S
  mark all the edges of $\mathcal{T}$
  initialize a stack containing all the edges of $\mathcal{T}$.
  while stack is non-empty **do**
    pop *ac* from stack and unmark it
    if *ac* is illegal **then**
      flip *ac* to *bd*
      for $xy \in \{ab, bc, cd, da\}$ **do**
        if *xy* is not marked **then**
          mark *xy* and push it on stack
  return $\mathcal{T}$

Analysis

- It is not obvious that this program halts!
- But in fact, it runs in $\Theta(n^2)$ time.

Proof.

- Each time we flip an edge, the lifted triangulation gets lower.
- So an edge can be flipped only once: Afterward, it remains above the lifted triangulation.
- There are $O(n^2)$ possible edges.
- So the algorithm runs in $O(n^2)$ time.
- Lower bound left as an exercise.

$\square$